

SPACECRAFT POTENTIAL CONTROL

**M.J. Mandell
V.A. Davis
B.M. Gardner
J.M. Hilton
I. Katz**

**Maxwell Technologies, Inc.
9210 Sky Park Court
San Diego, CA 92123-4302**

February 2001

Scientific Report No. 1

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.



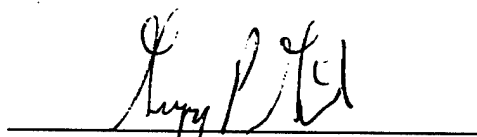
**AIR FORCE RESEARCH LABORATORY
Space Vehicles Directorate
29 Randolph Rd
AIR FORCE MATERIEL COMMAND
Hanscom AFB, MA 01731-3010**

20020702 046

“ This technical report has been reviewed and is approved for publication.”



DAVID L. COOKE
Contract Manager



GREGORY P. GINET, Chief
Space Weather Center of Excellence

This report has been reviewed by the ESC Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS).

Qualified requestors may obtain additional copies from the Defense Technical Information Center (DTIC). All others should apply to the National Technical Information Service (NTIS).

If your address has changed, if you wish to be removed from the mailing list, or if the addressee is no longer employed by your organization, please notify PL/IM, 29 Randolph Road, Hanscom AFB, MA. 01731-3010. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE February 2001	3. REPORT TYPE AND DATES COVERED Scientific Report #1 (08/25/98 - 03/01/01)		
4. TITLE AND SUBTITLE Spacecraft Potential Control		5. FUNDING NUMBERS Contract No. F19628-98-C-0074 PE:63410F PR: 2822 TA: GC WU:MX		
6. AUTHORS M. J. Mandell, V.A. Davis, B. Gardner, J.M. Hilton, I. Katz				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Maxwell Technologies, Systems Division, Inc. 9210 Sky Park Court San Diego, CA 92123		8. PERFORMING ORGANIZATION REPORT NUMBER MSD-DTR-01-16660		
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory 29 Randolph Road Hanscom Air Force Base, MA 01731		10. SPONSORING / MONITORING AGENCY REPORT NUMBER AFRL-VS-TR-2001-1619		
11. SUPPLEMENTARY NOTES Contract Manager: David L. Cooke/VSBXT				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for Public Release Distribution Unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) The objective of this contract is to develop improved numeric algorithms for the computation of spacecraft charging on earth orbiting spacecraft. This work is part of the Nascap2K program, which is a joint program with the Space Environment Effects (SEE) program at NASA/MSFC. The end result of the program will be a set of user friendly computer codes that compute spacecraft charging in all environments. This interim report describes the development of a computer code that computes spacecraft charging using the Boundary Element Method (BEM) and the development of the Object Toolkit, which is used to define spacecraft geometry and materials for Nascap2K. The algorithms contained in the BEM charging are described here. Nascap2K was tested and expanded by doing spacecraft charging calculations in support of the STEREO mission. A description of these calculations is included. Finally, this report describes a prototype Rapid Alert Charging Tool that uses geosynchronous environmental information to predict spacecraft charging.				
14. SUBJECT TERMS Nascap2K, Spacecraft charging, Boundary element method, Object toolkit, STEREO, Numeric algorithms			15. NUMBER OF PAGES 42 16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT SAR	

Table of Contents

1.	INTRODUCTION	1
2.	BEM CHARGING MODULE	4
	BOUNDARY ELEMENT METHOD ALGORITHM	4
	<i>Doing the Integrals</i>	5
	<i>Test for Accuracy</i>	6
	<i>Charging Equations</i>	7
	IMPLEMENTATION IN NASCAP-2K	12
	BEMDRIVER DETAILS	15
	BEMDRIVER COMMANDS	16
3.	OBJECT TOOLKIT	19
4.	PHOTOEMISSION	22
5.	STEREO ELECTROSTATIC ANALYSIS: INITIAL RESULTS	24
6.	RAPID ALERT CHARGING TOOL	39
7.	REFERENCES	42

1. Introduction

The objective of this contract is to develop improved numeric algorithms for the computation of spacecraft charging on earth orbiting spacecraft. This work is part of the Nascap-2K program, which is a joint program with the Space Environment Effects (SEE) program at NASA/MSFC. The end result of the program will be a set of user friendly computer codes that compute spacecraft charging in all environments. We envision the completed Nascap-2K software as a replacement for NASCAP/GEO, POLAR, NASCAP/LEO, and DynaPAC. The SEE program is supporting development of the graphical user interface.

The primary focus of work on this contract has been the development of a computer code that computes spacecraft charging using the Boundary Element Method (BEM) and on the development of the *Object Toolkit*, which is used to define spacecraft geometry and materials for Nascap-2K. The algorithms contained in the BEM charging module are described in Section 2. The *Object Toolkit* is described in Section 3.

The Nascap-2K computer codes are an extension of the DynaPAC (Dynamic Plasma Analysis Code) computer codes previously developed for the Air Force. DynaPAC was designed as a user-friendly and programmer-friendly workbench for studying plasma interactions with realistic spacecraft three dimensions. It enables plasma interactions specialist to perform realistic analyses with direct application to engineering problems.

DynaPAC was written in FORTRAN, with some support routines in C, for the UNIX operating system. Nascap-2K presently runs on the Win32 platform. A first effort under this contract was porting DynaPAC to the Win32 platform and the Developer Studio development environment. New written for Nascap-2K is being written in Java and C++. Under this contract DynaPAC was ported to the Linux operating system. We anticipate that porting the rest of Nascap-2K to Linux to be straightforward. Whether this will be done under this contract for a future one will be determined in consultation with the contract manager.

Nascap-2K consists of several modules. The modules communicate by XML files, direct subroutine calls (DLL import/export), JNI subroutine calls, and the DynaPAC database. XML files are used for input, which a user may wish to edit manually with a text editor or XML editor. XML Schemas specify the formats of the XML files. On Windows, the C++ modules validate their input files in accordance with the Schemas. Modules which may be executed indirectly (*BEMDriver* and all the DLL's) need to be placed where the system can find them (i.e., some directory in the user's path).

The following modules have been developed for Nascap-2K:

1. *Object Toolkit* (Java Application) is used to create finite-element representations of spacecraft surfaces. It also has materials editing capability, and can import Patran objects. It can also import objects from the DynaPAC database. Output (in XML) contains the recipe for recreating/reassembling the object, object definition by nodes and elements, and material definitions. *Object Toolkit* is described in Section 3.

2. *Nascap2K GUI* (Java Application) is the user interface for Nascap-2K. It is based on an index-tab metaphor, and contains tabs for problem selection, object definition (with a limited version of *Object Toolkit*), initial conditions (not implemented), environment specification, runscript creation and execution (rudimentary), "TermTalk-like" results analysis, and 3D display of surface potentials and fields. In addition to the object definition output file, it writes the run directives for *BEMDriver* (XML), the Windows Script file to run *BEMDriver* (XML/JScript), and a "project file" (XML) to save its state. This module was developed under contract with NASA and is not described further here.
3. *BEMDriver* (C++ Windows Console Executable.) (A Windows Console Executable is an executable that is run by typing the executable name in a console window, similar to the way the DynaPAC modules are run under UNIX.) *BEMDriver* orchestrates the running of a simulation. It reads an XML input file containing the commands to be executed, and passes these commands on to the *BEMDLL*. Additional details appear in Section 2.
4. *BEMDLL* (C++ Windows DLL.) (A Windows DLL is the same as a UNIX .so, shared object) *BEMDLL* performs the Boundary Element Method analysis. It reads the object definition output file (XML), converts the object information to the DynaPAC structure, and stores it in the database. It exports standard methods (called by *BEMDriver* to orchestrate calculations) and JNI methods (called by the Java Applications to retrieve results). Discussion of the algorithms used is in Section 2 below.
5. *DynaBase* (Fortran Windows DLL) is the C++ callable gateway to the DynaPAC database.
6. *Lapack* (C++ Windows DLL) is a custom implementation of matrix solver/inverter programs needed by Nascap-2K.
7. *Photoemission* (Java Application) may be used to build the XML photo-emission spectrum files needed for solar wind environments. Details appear in Section 4.

Under direction of contract manager Dr. David Cooke, we supported the STEREO mission by doing spacecraft charging calculations. This provided an opportunity to test and expand the capabilities of the Nascap-2K computer code and its algorithms. This work is described in a presentation made at the STEREO/Impact SWG Meeting in Berkeley, CA. This presentation is included here as Section 5 below.

In addition, this contract supported development of a Rapid Alert Charging Tool. This work is described in Section 6 below and in the publication I. Katz, V. A. Davis, M. J. Mandell, D. L. Cooke, R. Hilmer, L. Habash Krause, *Forecasting Satellite Charging: Combining Space Weather and Spacecraft Charging*, AIAA 2000-0369.

The scientists and other researchers who contributed to this work are as follows: Dr. Myron. J. Mandell, Dr. Ira Katz, Mr. Jeffery M. Hilton, Dr. Victoria A. Davis, Mr. David Monjo, Mr. Dale Lovell (summer intern), and Ms. Barbara M. Gardner.

This contract is a follow on to work performed under earlier contracts F19628-91-C-0187, Space System-Environment Interactions Investigation, F19628-93-C-0050 Modeling and Post Mission Data Analysis, and F19628-89-C-0032 Analysis of Dynamical Plasma Interactions with High Voltage Spacecraft. NASA supports related work under contract NAS8-98220.

The following publications were supported in total or in part by this contract.

I. Katz, V. A. Davis, M. J. Mandell, D. L. Cooke, R. Hilmer, L. Habash Krause, *Forecasting Satellite Charging: Combining Space Weather and Spacecraft Charging*, AIAA 2000-0369.

M. J. Mandell, I. Katz, D. L. Cooke, *Towards a More Robust Spacecraft Charging Algorithm*, AIAA 99-0379.

M. J. Mandell, I. Katz, J. M. Hilton, J. Minor, D. L. Cooke, *NASCAP-2K--A Spacecraft Charging Analysis Code for the 21st Century*, AIAA 2001-0957.

2. BEM Charging Module

Boundary Element Method Algorithm

Among the difficulties of developing accurate and robust algorithms for spacecraft charging has been the inability to calculate accurate electric fields, let alone to predict how electric fields will change as a result of surface potential changes. We proposed at the AFRL Spacecraft Charging Conference (November 1998) and at the AIAA Aerospace Sciences Meeting (January 1999) using the Boundary Element Method to calculate accurate electric fields, and as the basis for implicit charging equations.

The Boundary Element Method (BEM)¹ is a means for relating fields and potentials in a region to sources on the boundary of the region. It is comparable to a sum over the coulomb field of all the charges in a region rather than a field solution. In our case, the region is the space exterior to a spacecraft, and the boundary is the spacecraft surface. Also, we assume the "free space Green's function", *i.e.*, the potentials in the region will obey Laplace's equation.

The sources are sheets of charge coincident with the spacecraft model's surface elements. We assume that each surface element, j , has a constant charge density, σ_j . The familiar relation for the potential of a point charge then generalizes to an integral over the object surface:

$$V = \frac{q}{4\pi\epsilon_0 r} \quad \longrightarrow \quad (4\pi\epsilon_0)V_i = \sum_j \int d^2r_j \frac{\sigma_j}{r_{ij}}$$

where V_i which could be the potential at any point in space, is considered as the potential at the center of a surface element. Similarly, the familiar relation for the electric field of a point charge generalizes to:

$$E = \frac{q}{4\pi\epsilon_0 r^2} \quad \longrightarrow \quad (4\pi\epsilon_0)\mathbf{E}_i = \sum_j \int d^2r_j \frac{\sigma_j}{r_{ij}^3} (\mathbf{r}_i - \mathbf{r}_j)$$

where E_i is the electric field at some point in space, the point again taken at the center of a surface element.

We express the relations of potential and electric field to charge density as matrices:

$$V_i = [\mathbf{C}^{-1}]_{ij} \sigma_j \quad \mathbf{E}_i \cdot \mathbf{n}_i = \mathbf{F}_{ij} \sigma_j$$

These can be combined to obtain a relation between normal electric field and voltage:

$$\mathbf{E}_i \cdot \mathbf{n}_i = \mathbf{F}_{ik} \mathbf{C}_{kj} V_j$$

This last relation is the key to developing relations between surface charge, surface potential, and surface currents in order to derive charging equations.

Doing the Integrals

To get C^{-1} we need to do integrals of the form

$$\int d^2 r_j \frac{\sigma_j}{r_{ij}}$$

There are tricks to doing these integrals, which can be found in the literature. Denote the field point, r_i as P , and take the domain of integration as triangle ABC . Then, the vector from the field point to anywhere on the triangle can be parameterized as

$$\mathbf{r}_{ij} = \overline{PA} + u\overline{AB} + uv\overline{BC}$$

where PA , AB , and BC are vectors between pairs of points, and u and v are parameters, each in the interval $[0,1]$. The square of the distance, r_{ij} , then becomes

$$r_{ij}^2 \equiv \mathbf{r}_{ij} \cdot \mathbf{r}_{ij} = \sum_{0 \leq k, l \leq 2} T_{kl} u^k v^l$$

where the coefficients T_{kl} are pairwise scalar products of the three vectors above. Now, the integral can be expressed as

$$\int d^2 r_j \frac{\sigma_j}{r_{ij}} = \int_0^1 dv \int_0^1 u du \frac{\sigma_j}{\sqrt{V_0 + V_1 u + V_2 u^2}}$$

where the V_i coefficients are functions of v .

The inner integral (over u) may be found in standard integral tables²:

$$\begin{aligned} \int x dx \frac{1}{\sqrt{c + bx + ax^2}} &= \frac{\sqrt{c + bx + ax^2}}{a} - \frac{b}{2a} \int dx \frac{1}{\sqrt{c + bx + ax^2}} \\ \int dx \frac{1}{\sqrt{c + bx + ax^2}} &= \frac{1}{\sqrt{a}} \log \left| 2\sqrt{a}\sqrt{c + bx + ax^2} + 2ax + b \right| \end{aligned}$$

We are left to do the outer integral (over v) numerically. To facilitate this, we select the vertex A such that the scalar product $PA \cdot BC$ is the minimum of the three choices. Then, very few integration points are needed in the outer integral (over v). (We use 5-point Simpson's rule in the current implementation.)

The integrals for the electric field are similar, although there are more of them. The same techniques apply.

Test for Accuracy

We test for accuracy by calculating the electric fields on the surface elements of a uniform potential sphere. The sphere model (see Figure 1) was originally defined using Patran, and has 90 surface elements and 92 nodes. It provides a fairly coarse mesh over the surface of the sphere.

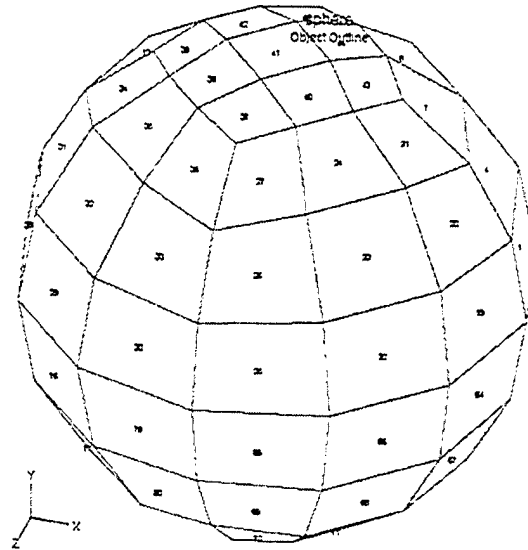


Figure 1. Patran sphere.

Figure 2 shows the result of the BEM calculation. Note that the total variation of electric field over the surface is only three percent. Also, the apparent radius of the sphere (given by V/E) is within three percent of the nominal radius.

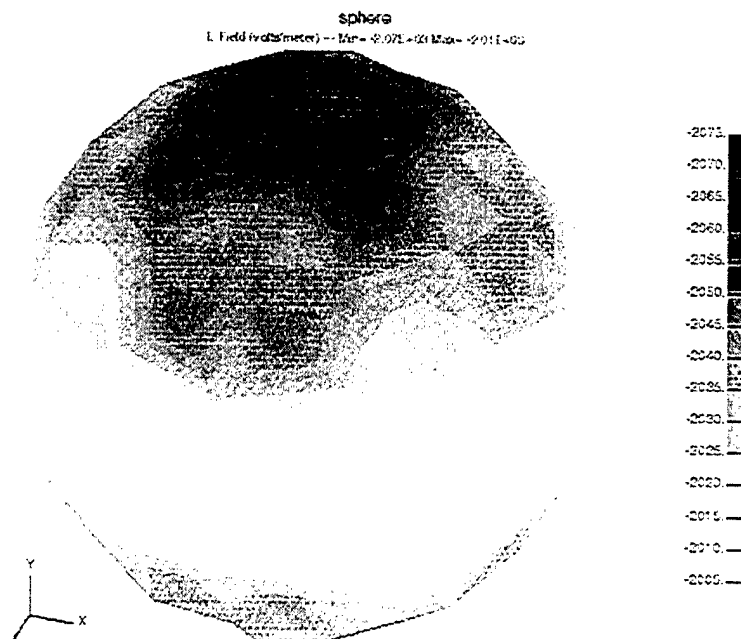


Figure 2. Normal electric field on a uniform potential sphere as computed using BEM.

By contrast, the same calculation using DynaPAC gives an electric field variation of seven percent, and a mean value of electric field that is about fifteen percent high. NASCAP/LEO gives an eight percent variation, and a value that is about twenty-five percent high. No comparison can be made with NASCAP/GEO, as that code neither calculates external electric fields, nor can it accept a sphere as geometrical input.

Charging Equations

To develop charging equations, we need to express physical charges on physical surfaces in terms of the voltages on the same objects. We can then proceed to calculate what voltage changes will be produced by changes in charge (currents). Because the interior of a spacecraft is NOT free space, the physical charge densities bear no relation to the σ 's, which we have now eliminated from the calculation.

To get the physical charges we use Maxwell's divergence equation in the form $\sigma = \nabla \cdot \mathbf{D}$. Figure 3 shows the "Gaussian pillboxes" used to calculate the actual surface charges on insulating surfaces and on conductors.

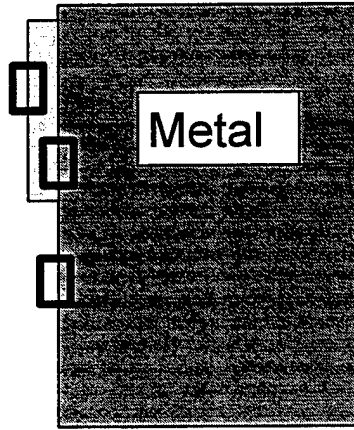


Figure 3. "Gaussian pillboxes" used to calculate the actual surface charges on insulating surfaces and on conductors.

For an insulating surface, the external field is $\mathbf{E} \cdot \mathbf{n}$, which we obtain from the BEM solution. The internal field is related to the capacitance between the insulating surface and its underlying conductor. Thus, the total charge, q_i on such a surface is given by:

$$q_i = A_i(\mathbf{E} \cdot \mathbf{n})_i + C_{ic}(V_i - V_c)$$

For a conductor, since charges are mobile, it is not useful to know the charge on each individual surface, but we need to work with the total charge on the conductor. Conducting surfaces include the obvious "bare" surfaces, as well as the surfaces that underlie the insulating surfaces. In both cases, we have zero electric field internal to the metal. To obtain the total charge on all the surfaces of the conductor, we need to sum the external-field charge terms for the bare conducting surfaces, plus the capacitive-charge terms for the insulator-metal interfaces:

$$Q_c = \sum_{bare} A_i(\mathbf{E} \cdot \mathbf{n})_i - \sum_{insulators} C_{ic}(V_i - V_c)$$

We previously found the BEM expression for the external fields in terms of the cell potentials:

$$\mathbf{E}_i \cdot \mathbf{n}_i = \mathbf{F}_{ik} \mathbf{C}_{kj} V_j$$

Substituting this into the equations for q_i and Q_c , performing the indicated sums, and adding the capacitive terms, we get the matrix equation:

$$\mathbf{Q} = \mathbf{G}\mathbf{V}$$

where the vectors are composed of contributions from all the insulating surfaces and a single term representing all the bare conducting surfaces.

$$\begin{aligned} \mathbf{Q} &= \{q_1, q_2, \dots, q_n, Q_c\} \\ \mathbf{V} &= \{V_1, V_2, \dots, V_n, V_c\} \end{aligned}$$

where \mathbf{Q} and \mathbf{V} only contain entries for those entities physically capable of accumulating charge, viz., conductors and insulating surfaces, and the charges and potentials are related by the charging matrix, \mathbf{G} .

We are looking for an equation that relates currents to voltage changes. So, we differentiate the charge equation in time:

$$\mathbf{I} \equiv \dot{\mathbf{Q}} = \mathbf{G}\dot{\mathbf{V}}$$

Discretize to a finite time interval:

$$\mathbf{I}\Delta t = \mathbf{G}[\mathbf{V}(t + \Delta t) - \mathbf{V}(t)]$$

Implicitize by evaluating current at the final time (also, for simplicity, changing $[t+\Delta t, t]$ to $[t, 0]$):

$$\mathbf{I}(t)t = \mathbf{G}[\mathbf{V}(t) - \mathbf{V}(0)]$$

Linearize currents with respect to voltage (since we do not know the final voltages at which the current is to be evaluated):

$$\mathbf{I}(t) \approx \mathbf{I}(0) + \mathbf{I}'[\mathbf{V}(t) - \mathbf{V}(0)]$$

And, solve the equation:

$$[\mathbf{G} - \mathbf{I}'t]\mathbf{V}(t) = [\mathbf{G} - \mathbf{I}'t]\mathbf{V}(0) + \mathbf{I}(0)t$$

This gives us a straightforward matrix equation. Everything on the right-hand-side is known, and we can solve using standard linear algebra equation solver packages.

Before proceeding to examples, it is worth commenting on the derivative of current, $\mathbf{I}'_{ij} = \partial \mathbf{I}_i / \partial V_j$. Consider three cases:

1. For current sources such as incident plasma current, we usually approximate the current as a function of the local voltage only. This gives a diagonal term in \mathbf{I}'_{ij} . Since such currents decay exponentially, some care must be taken not to underestimate the final current if the

voltage is changing in such a direction that the current is increasing (i.e., electron current for a surface whose potential is increasing (toward zero) from a large negative potential).

2. Conductivity current, such as $I_i = \sigma(V_i - V_c)$, contributes off-diagonal as well as diagonal terms to the current derivative matrix. Surface conductivity contributes many more off-diagonal terms.
3. The case that causes great difficulties for NASCAP/GEO is $I_i = I_f(E \cdot n)$. This occurs for a photo-emitting surface at negative potentials. The problem is that the local electric field is a function of the potentials on all the surfaces. But, the BEM provides us with exactly that function. So, we can now compute the term

$$I'_{ij} \equiv \partial I_i / \partial V_j = \partial I_i / \partial E_i \times \partial E_i / \partial V_j$$

using the relation

$$\mathbf{E}_i \cdot \mathbf{n}_i = \mathbf{F}_{ik} \mathbf{C}_{kj} V_j$$

derived from the BEM.

Example: Sunlit Sphere

We are now ready to recalculate the charging of a sunlit Teflon sphere in a 1 cm^{-3} , 20 keV plasma. The NASCAP/GEO version of this was published in 1978.³ A modern version of the original result is shown below in Figure 4. The dark side of the sphere gradually charges negative due to incident plasma electrons, while photoemission grounds the sunlit side. The sun direction is (1,1,1) (from the upper right). Eventually, however, the strong negative potentials due to the dark surfaces "wrap around" the sphere and form a barrier to photoelectron escape. The potential of each sunlit surface is subsequently determined by the condition that its electron field has a small, positive value (too small to be seen in Figure 4), so that just the right fraction of photoelectrons can escape over the barrier.

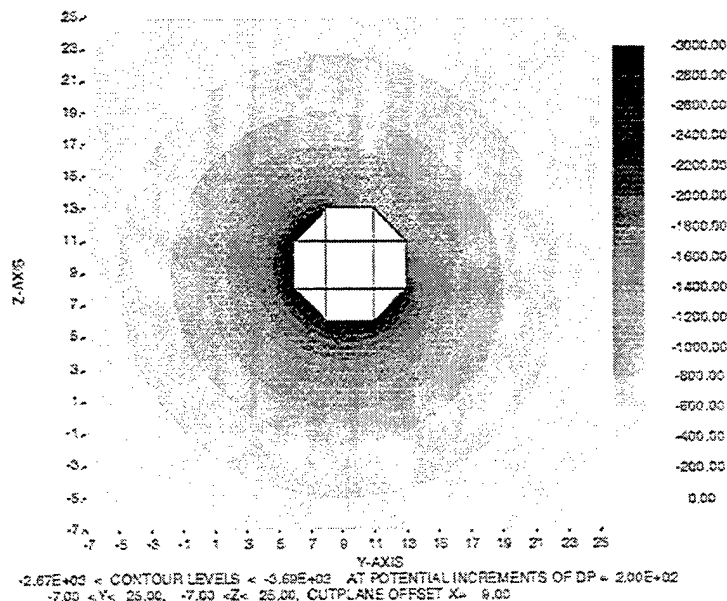


Figure 4. Potentials about sunlit QSphere in space as computed by NASCAP/GEO.

Figure 5 and Figure 6 below show the BEM solution (using DynaPAC graphics). The subsolar point is the least negative point.

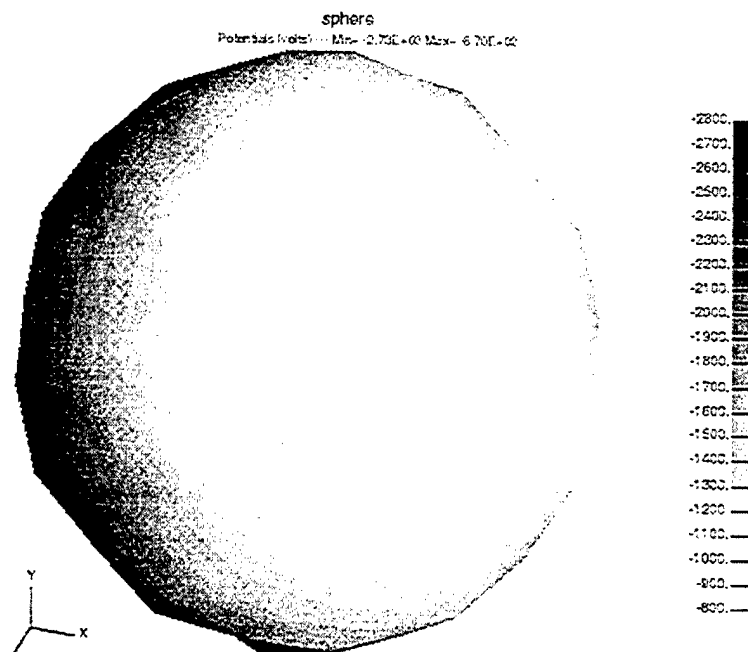


Figure 5. Potentials on sunlit Patran sphere in space viewed from direction (1,2,3) as computed by BEM.

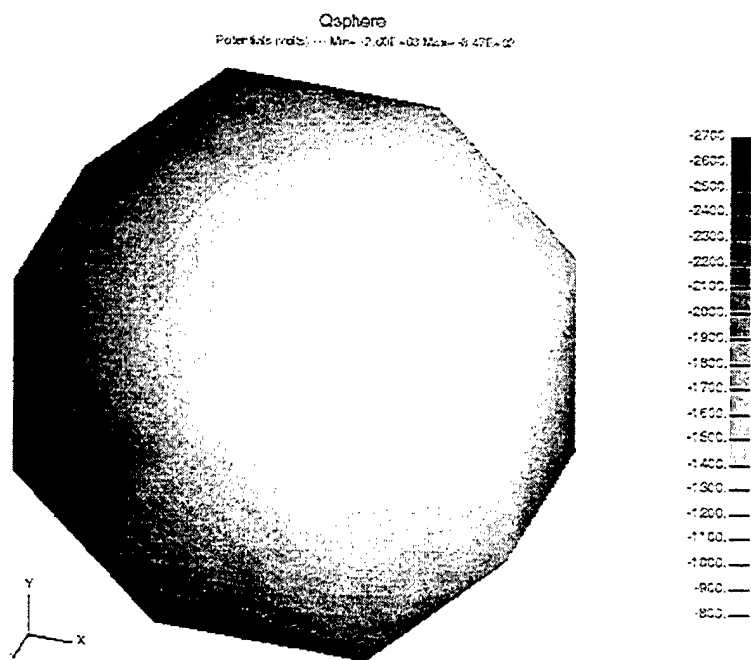


Figure 6. Potentials on sunlit QSphere in space viewed from direction (1,2,3) as computed by BEM.

Figure 7 is another view of the BEM solution, showing more of the dark side. Despite the fairly coarse gridding on the sphere, the gradual potential gradient on the sunlit side and the constant, large negative potential on the dark side are clearly seen in the BEM solution.

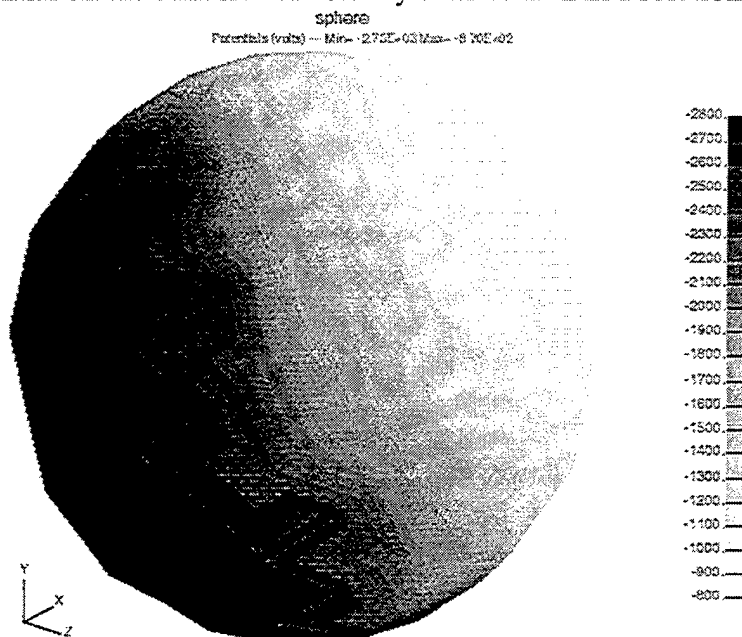


Figure 7. Potentials on sunlit Patran sphere in space as computed by BEM.

The table below compares the original NASCAP/GEO solution with the current NASCAP/GEO and Nascap-2K BEM solutions.

Table 1. Results comparison.

	NASCAP/GEO 1978	NASCAP/GEO 1999	BEM QSphere	BEM Sphere
Min. Potential	-3600	-2700	-2880	-2730
Max. Potential	-1000	-1000	-1160	-870
Min. Field			-18300	-5550
Max. Field			4.64	4.58

Note that, even though the same input is used, we can no longer reproduce the original NASCAP/GEO 1978 result because the treatments of secondary emission, backscatter, etc. have changed in unknown ways. But, the general character of the solution is the same, and differences among the remaining calculations are all plausible. In particular, the difference in minimum electric field (which occurs in the dark area near the terminator) is attributable to the smaller cell size for the QSphere compared with the Patran sphere. (Since we currently have no way of defining a QSphere for DynaPAC on the Win32 platform, the QSphere was manually input to the BEM module, and no graphics were available.) We did not keep careful track of the timestepping, but the case of all negative potentials (with photo-emission currents depending on electric field) seemed to go very smoothly.

Implementation in Nascap-2K

The BEM Charging Module is a DLL (BEMDLL.DLL) written in C++, and implemented on the Win32 platform. It uses another DLL, DynaBASE.DLL, to call DynaPAC's database routines, return geometric and material information to the caller, and insert the caller's results, surface potentials and electric fields, into the DynaPAC database. Nascap-2K can display time histories as plots or text and a three-dimensional representation of the surface potentials. Existing DynaPAC graphics routines can also be used to display results.

BEMDLL is accessed by using either the *BEMDriver* application described below or the *Nascap-2K GUI*, which is under development.

Formulations for incident plasma current, secondary emission, etc. have been adapted to C++ from the SEE Handbook, which contains Java versions of the original NASCAP/GEO FORTRAN routines. In principle, these should give the same results as NASCAP/GEO.

Treatments for shadowing and for a photo-emission spectrum were incorporated. Multiple biased conductors and a model for solar wind (beaming) ions were added.

This algorithm was also implemented in the SEE Spacecraft Charging Handbook.⁴ The implementation in the SEE Handbook was refined by testing against a variety of spacecraft models and environments. The accuracy and robustness of the algorithm was much improved, and the improvements were implemented in the BEM Charging Module. Figure 8 shows the default spacecraft geometry for the SEE Spacecraft Charging Handbook. The solar arrays are material Solar with Black Kapton backs. The sides of the body are covered with OSRs and everything else is Kapton. The plot shown in Figure 9 shows five minutes of charging for this object in sunlight using the NASA recommended Worst Case environment. The calculation was done using 20 geometrically progressive timesteps. Figure 9 shows potential vs. time for the chassis and the minimum and maximum insulator potentials. Detailed charging results appear in Figure 10. Figure 11 shows the first second of charging, illustrating the initial positive potentials. Figure 12 shows the first second of charging in eclipse. In the latter case, the model still initially charges positively because of the high emission of the solar cells. With only ten timesteps the calculation shows some wiggles. As shown in Figure 13, increasing the number of timesteps to twenty damps out the wiggles while leading to very nearly the same result.

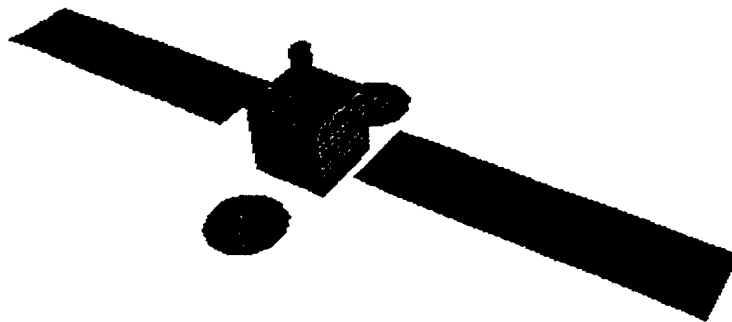


Figure 8. Default spacecraft in SEE Spacecraft Charging Handbook.

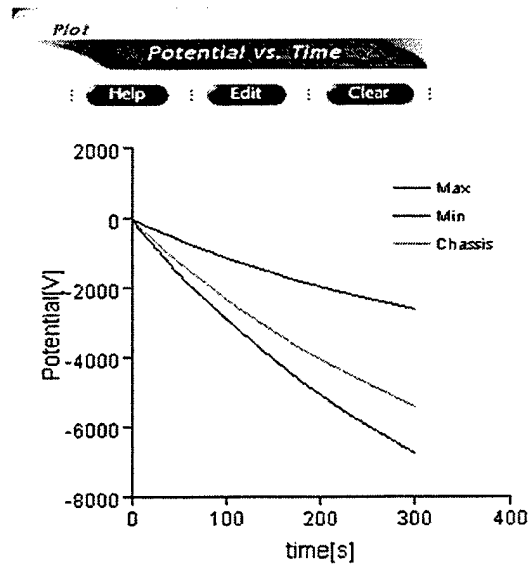


Figure 9. Five minutes of charging in sunlight.

Material:	Kapton	OSR	Solar Cells	Black Kapton
Color:	Red	Green	Blue	Yellow
Sunlit Area (m ²)	15.1	4.00	32.0	0
Dark Area (m ²)	15.5	4.00	0	32.0
Max. Potential (kV)	-4.400	-4.699	-2.612	-5.394
Min. Potential (kV)	-6.753	-5.414	-4.210	-5.394
Max. Differential (kV)	0.99352	0.69464	2.7818	0
Min. Differential (kV)	-1.3595	-0.019673	1.1843	0
Initial Current (A)	1.42e-4	3.07e-5	7.28e-4	-5.22e-5
Final Current (A)	-1.32e-5	-5.26e-7	4.39e-5	-3.09e-5

Figure 10. Results of five minutes of charging in sunlight. The colors refer to the SEE Spacecraft Charging representation of the spacecraft, which is shown in black and white in Figure 8.

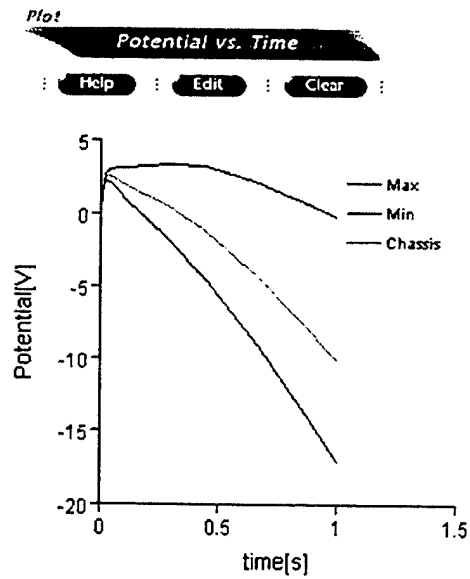


Figure 11. The first second of charging in sunlight.

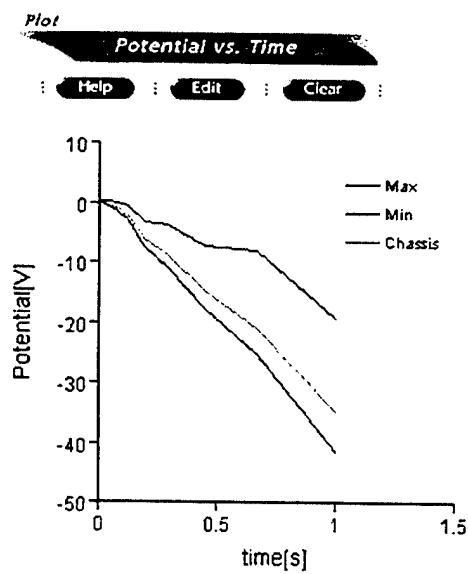


Figure 12. First second of charging in eclipse, 10 timesteps.

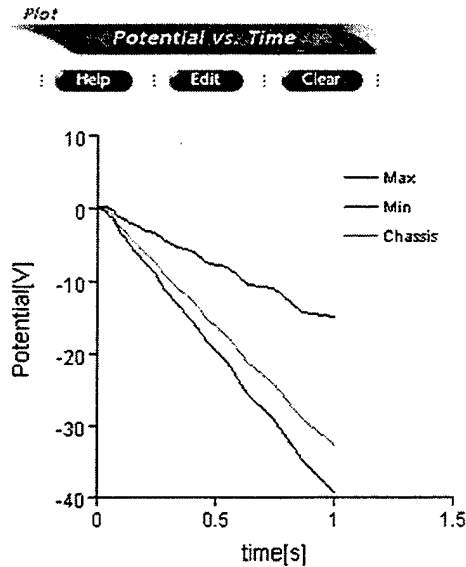


Figure 13. First second of charging in eclipse, 20 timesteps.

Results from this algorithm have also been compared to NASCAP/GEO calculations and to spreadsheet calculations, showing acceptable agreement.

BEMDriver Details

BEMDriver is a Windows console executable that orchestrates a charging analysis (performed by *BEMDLL*). It can be run from the command line or via a Windows Script file. The Windows Script file can be executed either directly or from the *Nascap-2K GUI*. (If the process is executed from the GUI under Windows 9x or Windows Me, *BEMDriver* output will appear in the Java Console window.) *BEMDriver* reads an XML command file specifying the details of the calculation, such as initialization, conductor potentials, and environment. The *Nascap-2K GUI* can be used to generate the Windows Script file and a rudimentary version of the XML command file. More sophisticated command files can be created using a text editor. It is planned that future versions of the *Nascap-2K GUI* will be able to automatically generate more complex command procedures.

BEMDriver takes as optional arguments the filename of the XML command file and the directory in which it is to run: *BEMDriver* [xmlcommandfile[rundirectory]]. If the rundirectory is omitted, it is assumed to be already in the correct directory. If the xmlcommandfile is omitted, it will present a dialog to find it.

A sample Windows Script file to run *BEMDriver* is:

```
<?xml version="1.0"?>

<job id="BEMDriver">
<script language="JScript">
  var Shell = WScript.CreateObject("WScript.Shell");
  Shell.Run("BEMDriver OnePanelDriver.xml
E:\\Nascap2000\\STEREO\\ ",1,true);
```

```
</script>
</job>
```

A sample xmlcommandfile (OnePanelDriver.xml) is:

```
<?xml version="1.0"?>

<BEM xmlns="x-schema:BEM_schema.xml" Prefix="OnePanel">
  <COMMAND cmd="ReadXML"
  FileName="E:\Nascap2000\STEREO\OnePanelObject.xml" />
  <COMMAND cmd="ReadDynaPAC" />
  <COMMAND cmd="Calculate_Matrices" />
  <COMMAND cmd="DefineInsulators" />
  <COMMAND cmd="SetSunDirection" x="0.0" y="0.0" z="1.0" />
  <COMMAND cmd="SetGeoEnvironment">
    <Environment nel="1000000.0" tel="3.0" nil="100000.0"
    til="1000.0" />
  </COMMAND>
  <COMMAND cmd="ReadPhotoemission" FileName="WindPhoto.xml"
  />
  <COMMAND cmd="SetConductorBias" Value="60." Index="2"
  Index2="1" />
  <COMMAND cmd="PrepareChargeMatrix" />
  <COMMAND cmd="DoTimeSteps">
    <TimeParams nsteps="90" endtime="180" begintime="0.0"
    mindt="0.1" />
  </COMMAND>
</BEM>
```

BEMDriver Commands

Currently implemented commands include:

ReadXML - reads object from XML info to Database

Attributes: FileName (optional)

- path to xml object file containing nodes and elements

ChildElements: None

ReadDynaPAC - Opens Database and gets object info

Attributes: Prefix (optional)

- prefix to use for Database files

ChildElements: None

ReadPhotoemission - Read photoelectron spectral data

Attributes: FileName (optional)

(if no FileName, looks for "prefix.photo.xml".

If that is not found, uses default photoemission.)

ChildElements: None

CalculateMatrices - Calculate the BEM matrices for the model.

Attributes: None.

ChildElements: None.

ReadPotentials - Read potentials from the Database.

Attributes: None.

ChildElements: None.

WritePotentials - Write potentials to the Database.

Attributes: None.

ChildElements: None.

DefineInsulators - Step towards eliminating conductive surfaces from the matrices.

Attributes: None.

ChildElements: None.

SetSolarWindIons- Ions beam from sun direction and are shadowed.

Maxwellian T11 is used as energy.

Attributes: None.

ChildElements: None.

SetConductorBias - Sets bias value for a conductor.

Attributes: Value - value of bias potential.

Index - index of biased conductor.

Index2 - index of reference conductor.

SetSunDirection - Sets direction from spacecraft to sun.

Attributes: x, y, and z - components of sun vector.

SetSunIntensity - Sets sun intensity

Attributes: Value - ratio to solar intensity at Earth's orbit.

SetGeoEnvironment - Sets (a sequence of) Geosynchronous Environments

ChildElements: Environment - (at least one required.)

Set_Element_Potentials - set potential of all elements to a single value.

Attributes: Value (defaults to 0.0)

PrepareChargeMatrix - Compute additional matrices needed for timestepping.

Attributes: None.

ChildElements: None.

DoTimeSteps - Perform timestepping of charging and potentials.

Attributes: None

ChildElements: TimeParams - (at least one required.)

Child Elements:

Environment - Geosynchronous Environment Description

Attributes: ne1, te1, ni1, ti1 (required) - properties of primary maxwellian;
ne2, te2, ni2, ti2 (optional) - properties of secondary maxwellian;
from, to (optional) - effective time for time-dependent environment

TimeParams - Parameters for setting timesteps

Attributes: (all optional)

begintime - timestamp (s) at beginning of timestep sequence
(default 0 or previous endtime).

endtime - timestamp (s) at end of timestep sequence (default 1).

nsteps - number of timesteps (default 1).

mindt - duration of first timestep (default 0.1).

3. Object Toolkit

Object Toolkit (Java Application) is used to create finite-element representations of spacecraft surfaces. It also has materials editing capability, and can import Patran objects. It can also import objects from the DynaPAC database. Output (in XML) contains the recipe for recreating/reassembling the object, object definition by nodes and elements, and material definitions. The Help Menu provides access to on-line documentation.

The screen for *Object Toolkit* looks as shown in Figure 14. The first four Cursor Tools are used to put the cursor in a mode to select specific objects, elements, edges, or nodes. The next two are used to put the cursor in a mode to translate or rotate the selected object. The final two Cursor Tools are used to put the cursor in a mode to translate or rotate the view. The Direct Movement and Rotation tools translate, scale, and rotate the view.

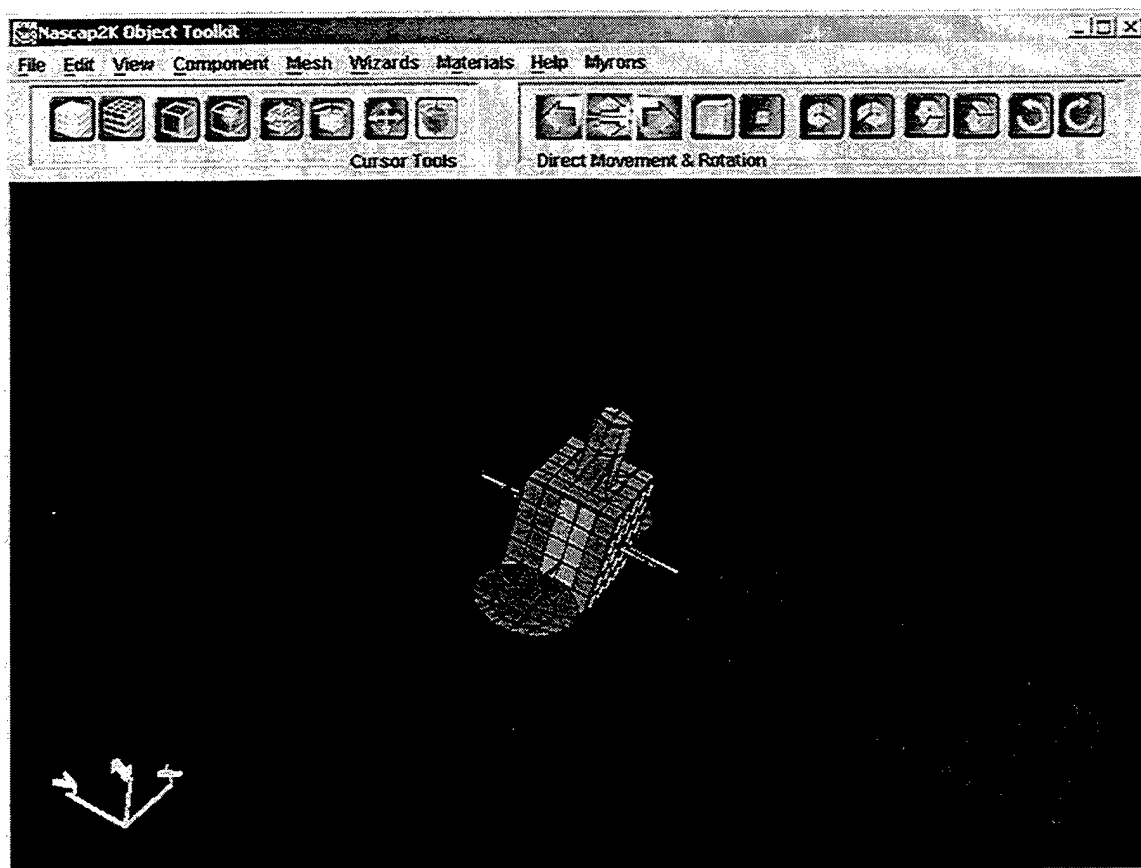


Figure 14. *Object Toolkit* screen.

Object Toolkit builds an object as a hierarchical assembly structure. Each "Assembly" consists of two "Components." Each "Component" may be an "Assembly" or a primitive object. The available primitive objects are:

- Brick – A hexahedron (six faces, eight nodes, twelve edges), which is usually rectangular.
- Boom – A subclass of Brick with thin dimensions (one zone) in the X and Y directions.

Panel – A subclass of Brick that is thin in the Z direction, and has the $\pm X$ and $\pm Y$ faces eliminated by sewing the edges together.

Dish – A primitive used to represent a parabolic antenna.

Cylinder – A primitive representing a 6-sided, 8-sided, or 4n-sided right cylinder (which may be tapered).

Primitive – An object defined by its mesh. Currently, a primitive can be created by converting a component, or by importing a Patran or DynaPAC object.

With each component of an assembly is associated a “Transformation” (from its local coordinates to the assembly’s coordinates) describing how it is to be positioned within the assembly. Components can be crudely aligned and positioned using the mouse tools, and precisely aligned and positioned with some of the menu items described below. Certain types of attachments (or “welds”) can be done (compatibly and without need for manual mesh editing) using “Wizards” provided as part of the tool.

The File menu includes “New”, “Open”, “Save”, “Save As”, “Save As BEM File”, “Import DynaPAC Object”, “Import SEE Handbook Materials File”, and Exit.

The Edit menu includes “Undo”, “Redo”, “Cut”, “Copy”, “Paste”, “Paste Special”, “Edit Component”, “Delete Component”, and “Select All”. The “Undo” and “Redo” are not fully implemented. Extreme caution must be exercised when Editing and Deleting components as elements may not match up in the same way after sizes of any components are changed. Mysteriously distorted elements and mysteriously changed materials on elements can be caused by elements no longer matching up as intended.

The View menu includes are “Hide”, “Hide Others”, “Show All”, “Material”, “Conductor”, and “Select View”. Select View has options “From X Axis”, “From Y Axis”, “From Z Axis”, options “From -X Axis”, “From- Y Axis”, “From- Z Axis”.

The Components menu includes “Add New”, “Add Component from File”, “Node Relative Move”, “Align Edge”, “Center at Origin”, “Enter Assembly”, “Leave Assembly”, “Top Assembly”, “Consolidate Components into Assembly”, and “Convert Component into Primitive”. “Node Relative Move” is used to translate an assembly so that a node is at a specified location. “Align Edge” is used to rotate an assembly so that an edge is oriented as specified. These commands can be used to locate and orient the object for use by other tools, such as *GridTool* or *Nascap-2K GUI*. The three “Assembly” commands are used to select components that have been consolidated into Assemblies. Once a component has been converted to a primitive, only mesh editing is available.

The Mesh menu includes “Materials”, “Conductors”, “Create Element”, “Delete Element”, “Divide Element”, “Combine Triangles”, “Reverse Elements”, “Edit Node”, “Combine Nodes”, and “Rebuild Mesh”. “Materials” and “Conductors” is used to change the material or conductor on the selected element. “Rebuild Mesh” does a comprehensive rebuilding of the mesh, including renumbering. The mesh is usually only partially rebuilt each time a change is made in the object. With experience, we hope to determine all those cases where additional rebuilding is necessary and automate it, making this menu choice unnecessary.

There are five wizards that move components so that they join. Components combined using a wizard are automatically joined into assemblies. The five wizards are "Element to Element", "Edge to Element", "Element to Edge", "Edge to Edge", and "Surface to Surface". After using the "Surface to Surface" wizard it is usually necessary to do some mesh editing as joining two surfaces is too complex for easy automation.

The materials menu permits creating, deleting, and editing materials. Editing of which material is on a specific surface is set when defining the component and by editing a specific surface using the Materials choice on the Mesh menu.

4. Photoemission

Spacecraft in the Solar Wind normally charge to positive potentials (a few volts to tens of volts) to balance emitted photoelectron current with a very low current of ambient electrons. Determining just how positive a spacecraft will charge requires more detailed knowledge of the photoelectron spectrum than is required for geosynchronous charging, in which potentials are negative and of much larger magnitude. Nascap-2K accepts definition of a photoelectron spectrum as a sequence of energy-value pairs, where the value indicates the fraction of the photoelectron spectrum lying above the energy. The definition is read from an external XML file. A different spectrum may be specified for each material. If no spectrum is specified, the photoelectron spectrum is assumed to be a 2 eV Maxwellian.

We have been provided with a single spectrum, based on data for the WIND spacecraft. The XML file specifying this data is:

```
<?xml version="1.0"?>

<Photoemission xmlns="x_schema:material_schema.xml">
  <Material xmlns="" Name="OSR" totalemission="2.e-05">
    <Spectrum energy="0.0" Value="1.0"/>
    <Spectrum energy="5.18" Value=".108"/>
    <Spectrum energy="5.93" Value=".0876"/>
    <Spectrum energy="7.26" Value=".0641"/>
    <Spectrum energy="9.41" Value=".0417"/>
    <Spectrum energy="12.8" Value=".0233"/>
    <Spectrum energy="18.3" Value=".0119"/>
    <Spectrum energy="27.25" Value=".00522"/>
    <Spectrum energy="41.77" Value=".00154"/>
    <Spectrum energy="65.24" Value="2.4e-4"/>
  </Material>
</Photoemission>
```

We built a Java Application, *Photoemission*, to construct this file. The application is shown below in Figure 14. The "Validate" operation assures that the energies are increasing and the fractions decreasing. The file can then be saved.

Photoemission Spectrum

Material: **OSR** Total Emission: **2.e-05** A m⁻²

Energy	Fraction of Spectrum Above
0.0	1.0
5.18	.0164
5.93	.0113
7.26	.00662
9.41	.00389
12.8	.00172
18.3	5.57e-4
27.25	2.24e-4
41.77	6.05e-5
65.24	9.39e-6
Energy 10	
Energy 11	
Energy 12	
Energy 13	
Energy 14	
Energy 15	
Energy 16	
Energy 17	
Energy 18	
Energy 19	

Validate **Save** **Cancel**

Figure 15. Photoemission screen.

5. STEREO Electrostatic Analysis: Initial Results

STEREO Electrostatic Analysis

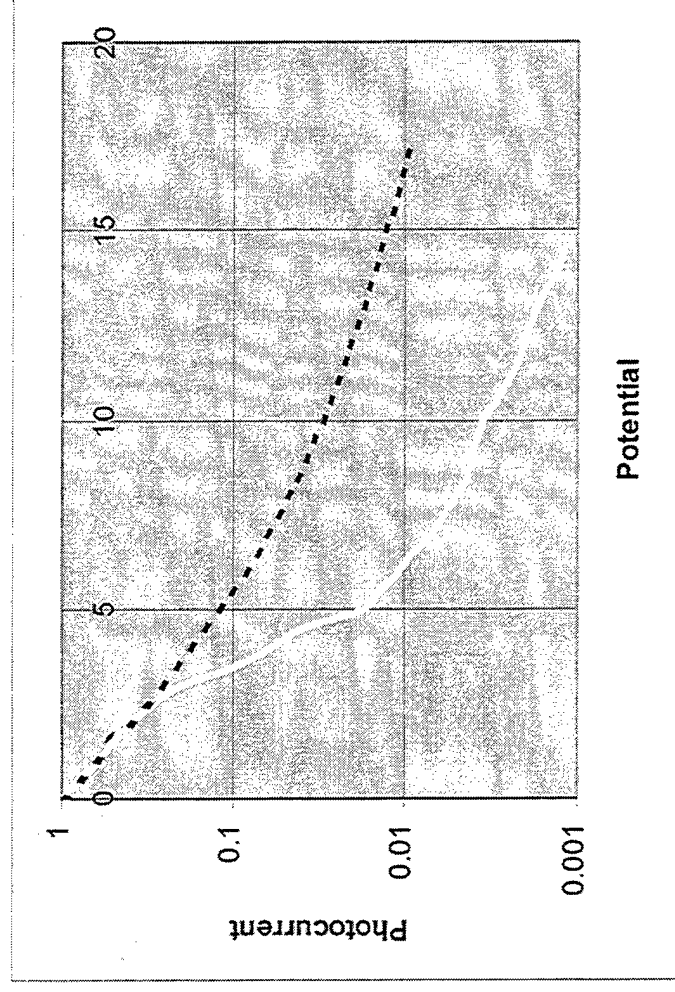
Initial Results

Myron J. Mandell
Ira Katz

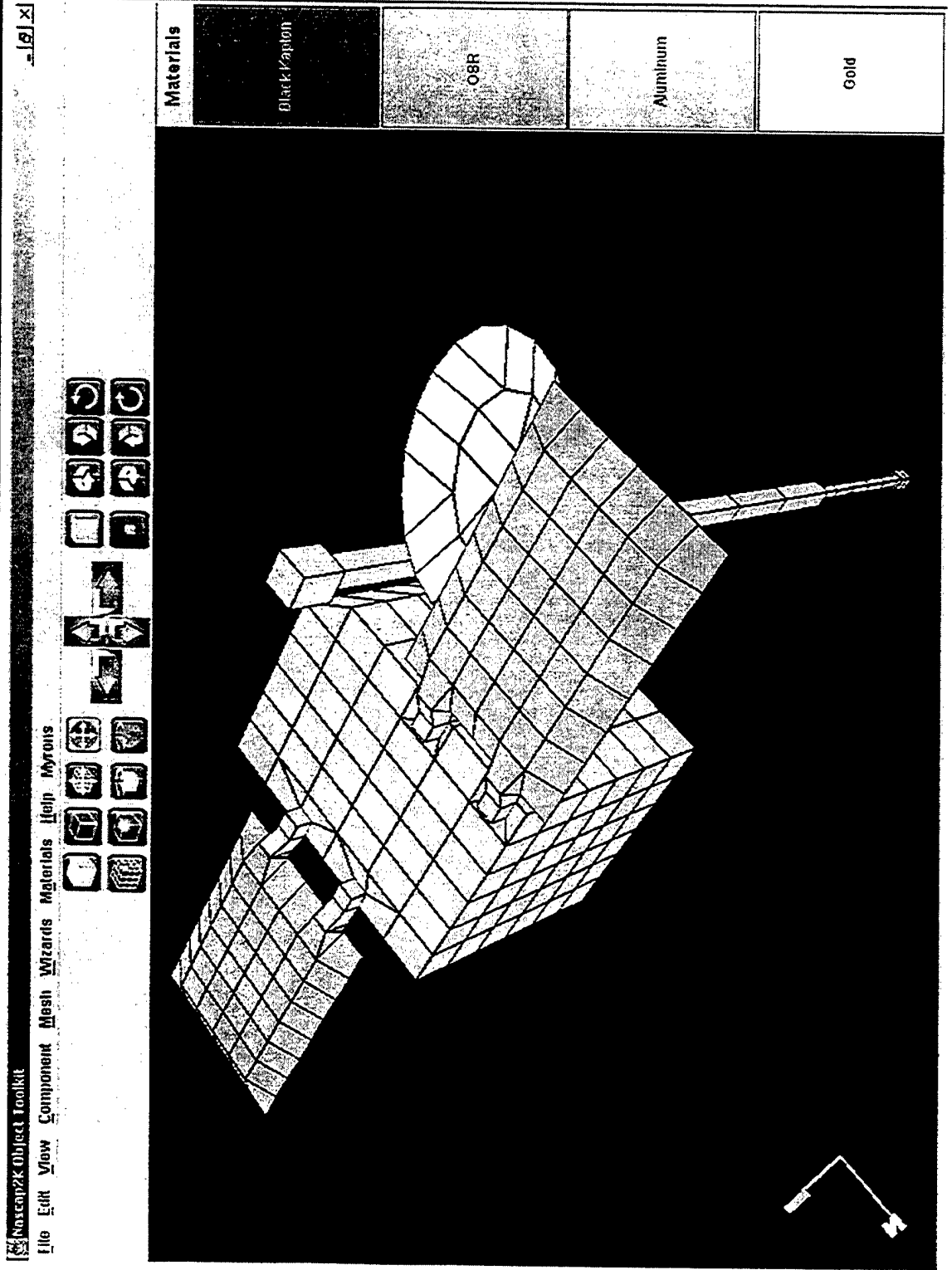
STEREO/Impact SWG Meeting
Berkeley, CA
December 12, 2000

- Solar Wind Charging Environment
- Nascap-2K Model for STEREO
- Overall and Differential Charging
- Conductivity of CMX Coverslips
- Nascap-2K Results

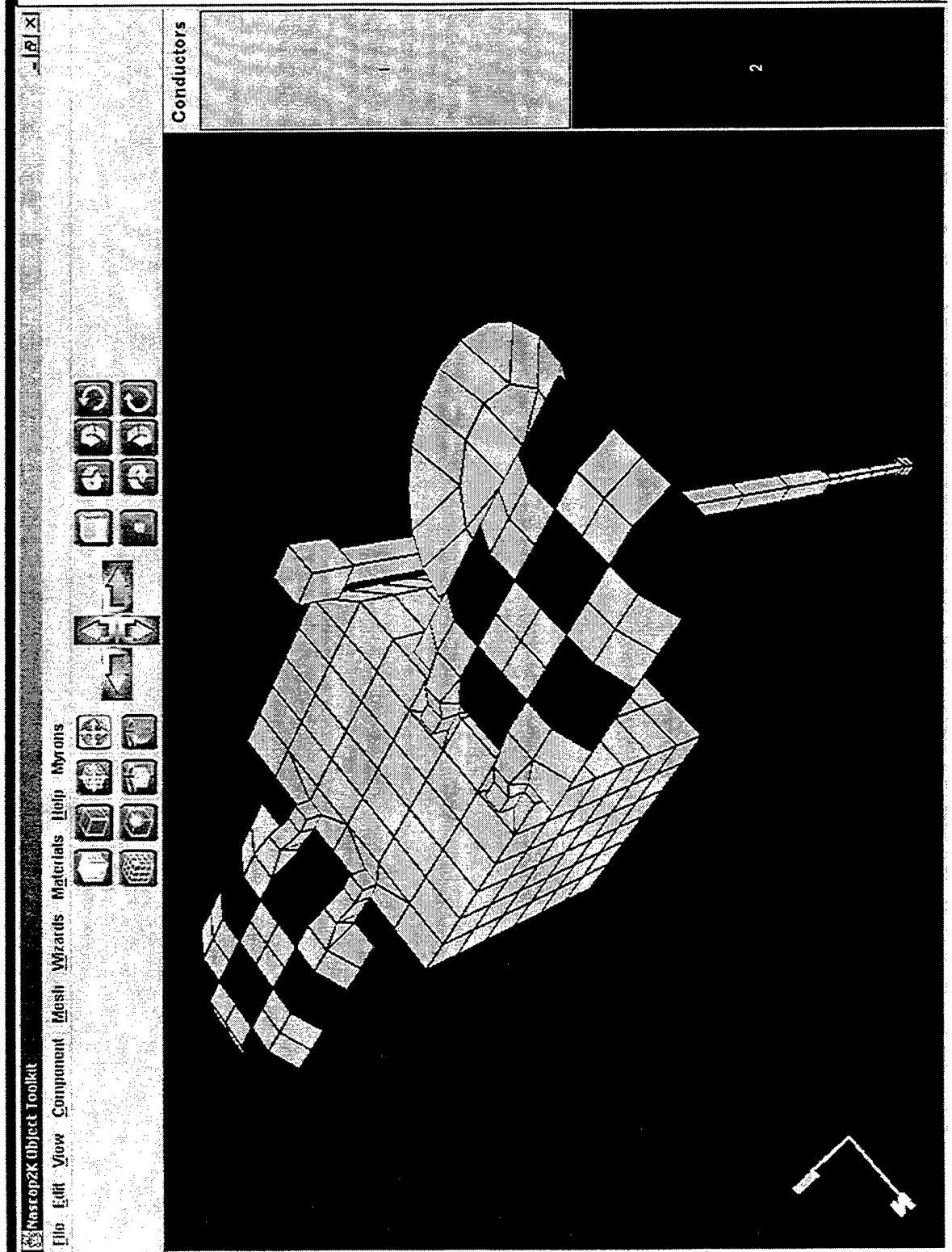
- Sunlight causes photoemission
Effective photoemission decreases
with potential
- Protons are beaming from sun
direction
 $v \sim 400 \text{ km/s}$
 $E \sim 800 \text{ eV}$
- Electrons are isotropic
Plasma Density $\sim 10^6 \text{ m}^{-3}$
Plasma Temperature $\sim 3 \text{ eV}$
Debye Length $\sim 10 \text{ m}$



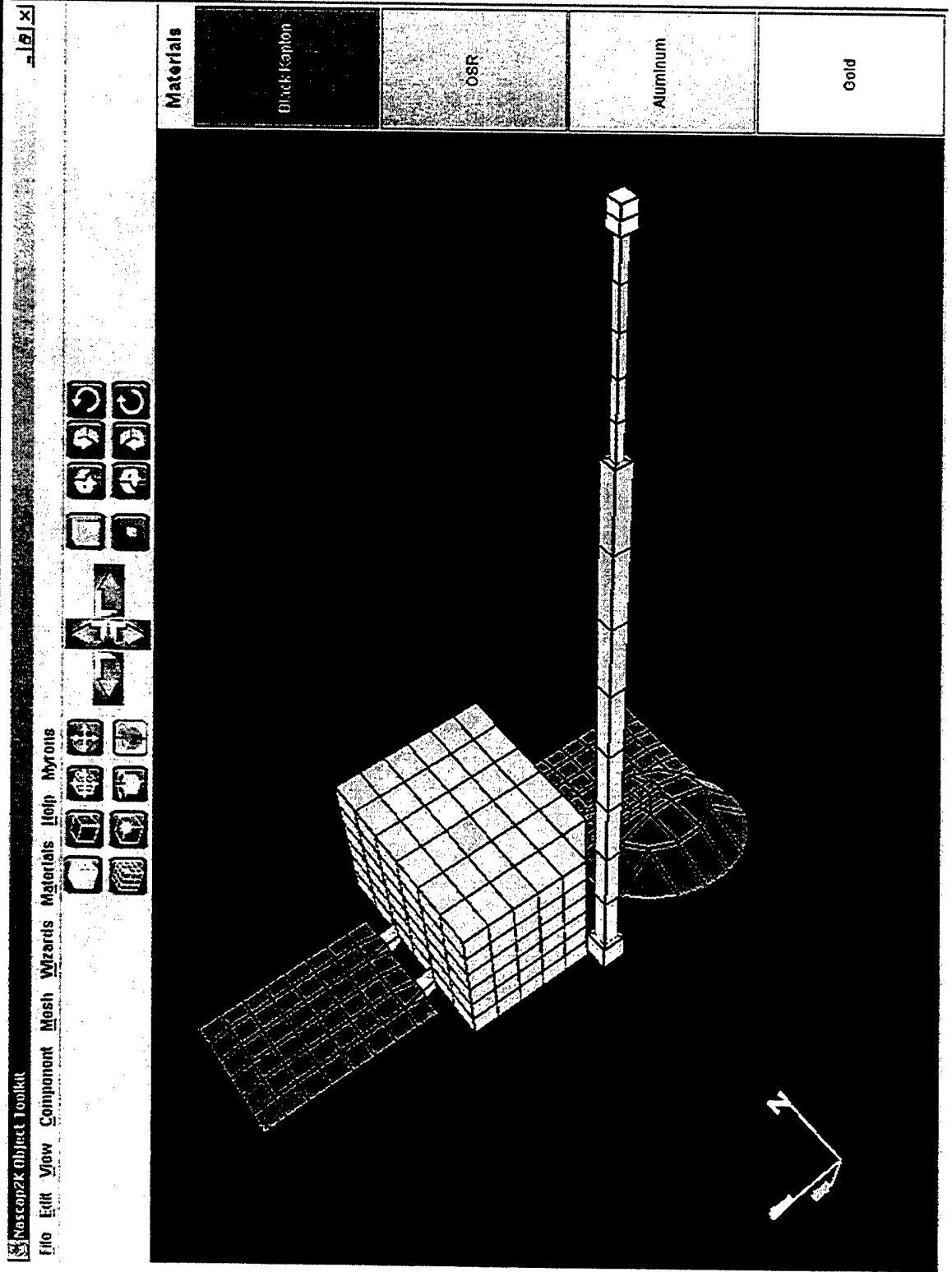
Nascap-2K Model



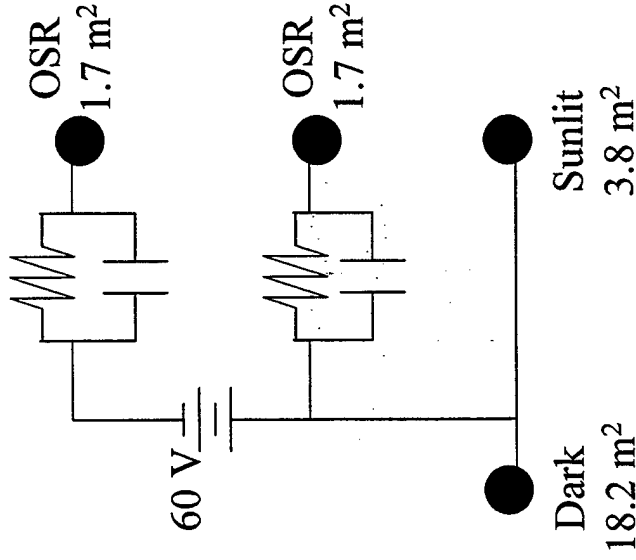
Nascap-2K Model



Nascap-2K Model



Circuit Analysis



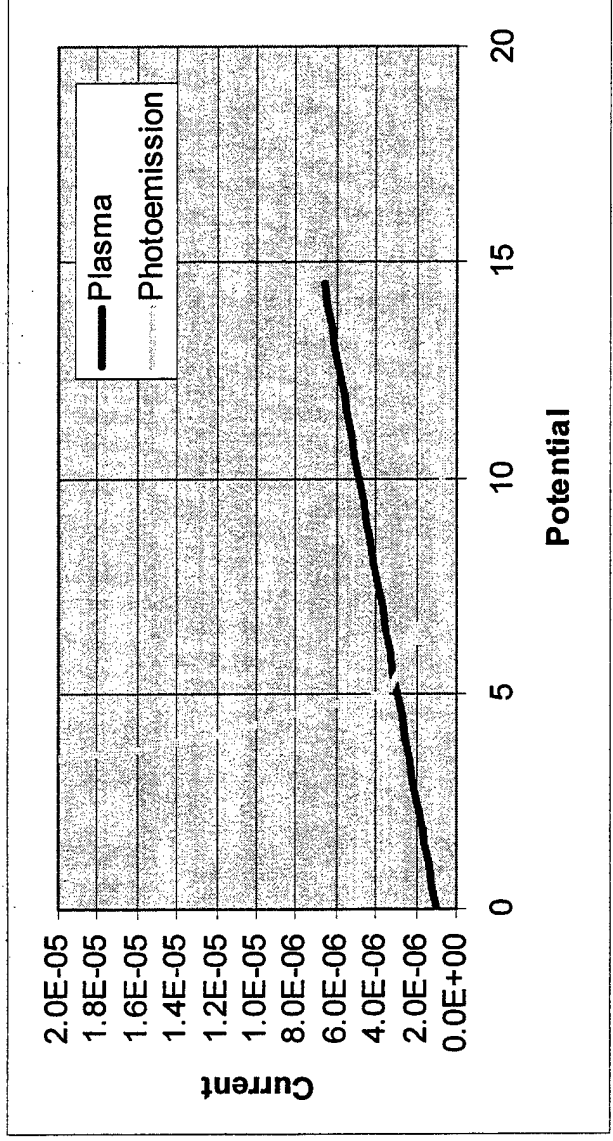
Overall Charging

$$\text{Electron Current} = A_{\text{total}} J_e (1 + V/T)$$

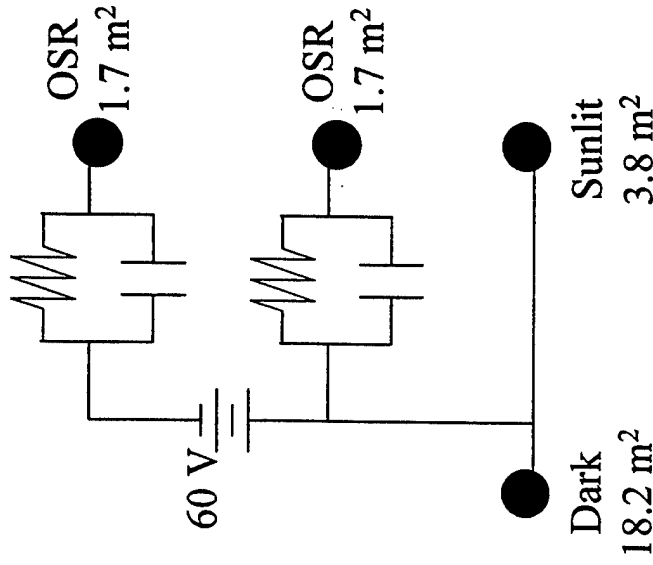
$$\text{Ion Current} = A_{\text{sunlit}} N_{\text{ev}}$$

$$\text{Photocurrent} = A_{\text{sunlit}} I_{\text{ph}}(0) f(V, E)$$

Capacitance is small, so s/c rapidly reaches potential for zero total current.



Circuit Analysis



Differential Charging

At uniform floating potential

Sunlit insulators have positive current

Dark insulators have negative current

Coverslips may have positive

conduction current

$$dV/dt = J/C$$

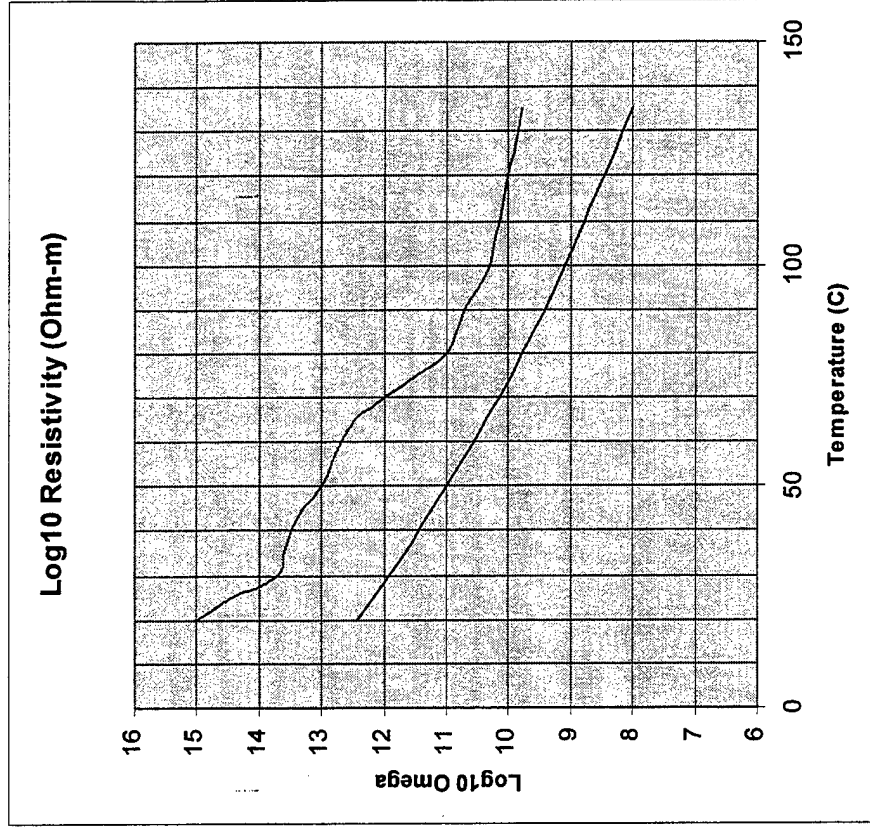
$$J \sim 3 \times 10^{-8} \text{ Am}^{-2}$$

$$C \sim 3 \times 10^{-7} \text{ Fm}^{-2}$$

$$dV/dt \sim 0.1 \text{ Vs}^{-1}$$

Charging rate will decrease with time

- Conductivity of CMX coverslips will bring some coverslips to elevated potential.
- We tend to believe higher of two data sources.
- Nominal resistivity taken as 10^{13} ohm-m
- $J = 60 \text{ V} \times 10^{-13} / 1.5 \times 10^{-4}$
- $J = 4 \times 10^{-8} \text{ A-m}^{-2}$ (comparable to environment currents)



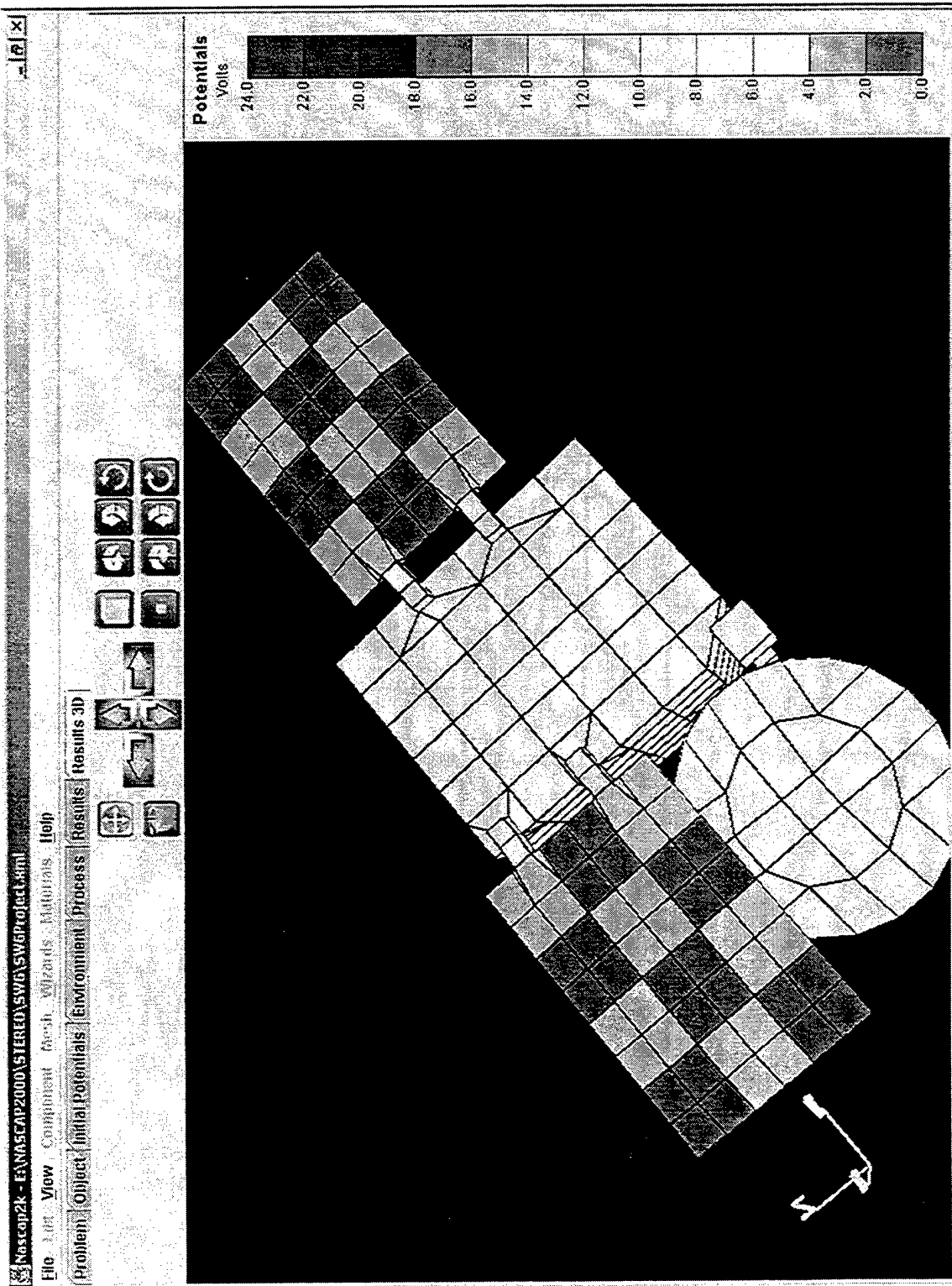
Environment:

$$n_e = 1 \times 10^6 \text{ m}^{-3}$$

$$E_i = 777 \text{ eV}$$

CMX Resistivity:	10^{17}	10^{13}	10^{12}	10^{13}
				$n_e = 1 \times 10^5 \text{ m}^{-3}$
OSR/60	8.1	9.0	20.18	23.5
OSR/0	8.0	7.9	7.55	14.4
Ground	4.4	4.3	4.2	8.0

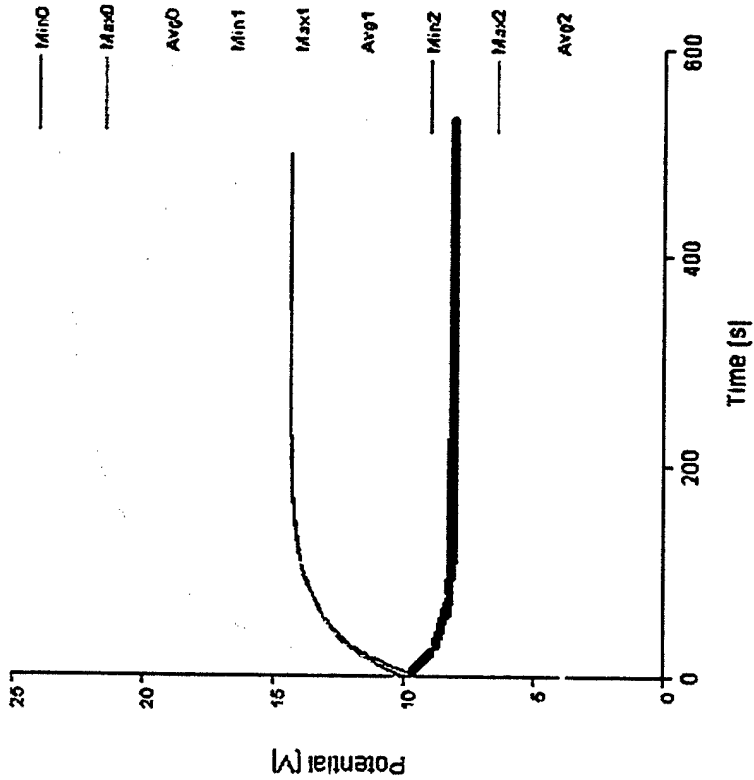
STEREO Potentials at Low Density



Low Density Charging History

Potential

Current Balance



$$\text{Conduction} = 45 \times 10^{-13} / 1.5 \times 10^{-4} = 3 \times 10^{-8}$$

$$\text{Electrons} = -4.6 \times 10^{-9} \times (1 + 24/3) = -4 \times 10^{-8}$$

$$\text{Protons} = 4 \times 10^5 \times 10^5 \times 1.6 \times 10^{-19} = 6.4 \times 10^{-9}$$

$$\text{Photocurrent} \sim 4 \times 10^{-9}$$

$$\text{Total} < 10^{-9} \text{ Am}^{-2}$$

- ITO Coatings are commonly grounded to local interconnects
- Surface potential variation reflects solar cell voltage pattern
- Positive cells will have enhanced electron collection, and may drive spacecraft negative

- Under normal conditions
 - Chassis is a few volts positive
 - Coverslips have a few volts positive differential
- Get some differential charging if
 - CMX is conductive (resistivity $< 10^{13}$ ohm-m) (or hot)
 - Solar wind becomes tenuous
- Small insulating patches in the dark can charge to -5 volts or more. Thus far, this does not seem to indicate a problem.
- ITO coating predicted to have little or no benefit, and may be harmful to mission.

6. Rapid Alert Charging Tool

The Rice University Magnetospheric Specification and Forecast Model (MSFM) is being used operationally to predict the magnetospheric electron and ion populations in the 100 eV to 200 keV energy range. These are the injected plasma sheet particles that form the ring current, and precisely those particles that cause spacecraft surface charging. In parallel, there have been improvements in the algorithms used to model charging of a given satellite in an assumed or measured environment, as well as advances in our understanding of what charge configurations make a spacecraft vulnerable to charging-related problems. This paper describes the first steps to applying the prediction of space weather to the prediction of spacecraft charging. The analysis predicts spacecraft charging by combining those parameters known to govern spacecraft charging with an environment description provided by MSFM. The first results are encouraging. Compared with three days of on-orbit spacecraft frame charging measurements from Charge Control System (CCS) flown on the Defense Satellite Communication System (DSCS) III B-7 satellite, the method predicted spacecraft charging on the two days when DSCS III charged the most ($V_f \geq 300$ V), but failed to predict charging on the third day when DSCS III experienced only mild charging ($V_f \leq 200$ V).

The three days were chosen so that there was no CCS plasma source operation and the satellite was not in eclipse. The three days, 45, 215, and 217 of 1996, also had spacecraft charging ranging from a peak of 200 V negative on day 217, to a peak of 600 V negative on day 215. Measured spacecraft potentials for the three days are shown in Figure 16. MSFM calculations of the electron and ion fluxes for eight logarithmically spaced bins for energies of 32 eV to 200 keV. Fluxes were calculated at intervals of 15 minutes.

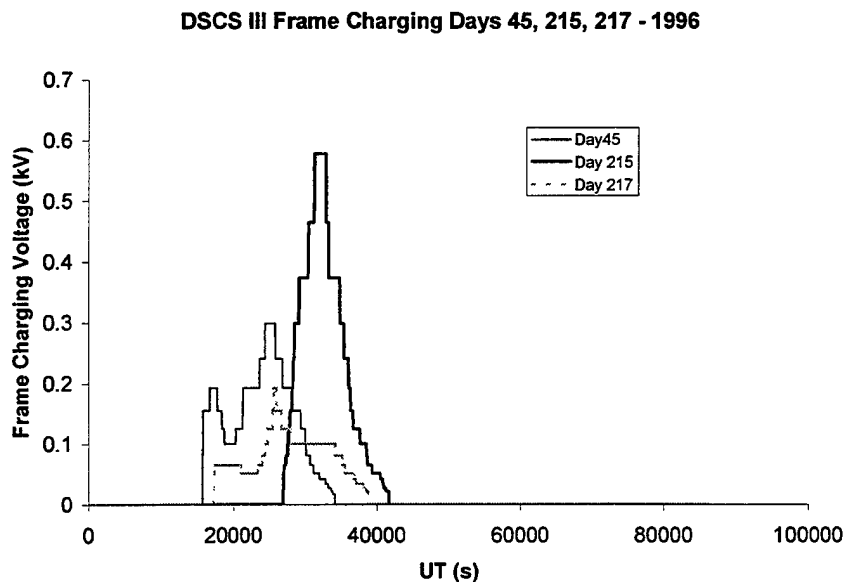


Figure 16. Flight data showing DSCS spacecraft charging for three days in 1996.

Net charging currents were calculated from the MSFM fluxes and material properties using the following formula.

$$I_t = \pi e \sum \Delta E_i (f_e(E_i)(1 - Y_{se}(E_i) - Y_{bse}(E_i)) - f_p(E_i)(1 - Y_{sp}(E_i)))$$

The values used for the secondary and backscatter yields are given in Table 2.

Table 2. Backscatter and secondary electron yields as a function of energy for incident ions and electrons.

E (keV)	0.03	0.10	0.32	1.00	3.16	10.00	31.62	100.00
Y_{se}	0.23	0.79	1.43	1.11	0.53	0.22	0.10	0.04
Y_{bse}	0.00	0.16	0.26	0.32	0.29	0.24	0.22	0.22
Y_{sp}	0.04	0.04	0.06	0.37	1.25	2.56	3.65	4.00

The negative of the net current density is shown in Figure 17. We expect the spacecraft to charge negatively when this current density is greater than zero. (Note the sign of the current was chosen to agree with the way the charging is shown in Figure 16.) Notice that the maximum calculated charging current occurs on Day 215 when DSCS III charged to -700 V. The calculated charging current density on Day 45 is very similar to that for Day 215, even though DSCS III only charged to -300 V on that day. No charging current is calculated for Day 217 when DSCS III charged to -200 V. The net charging current is never more than a few percent of the incident electron current due to secondary, backscatter, and ion currents.

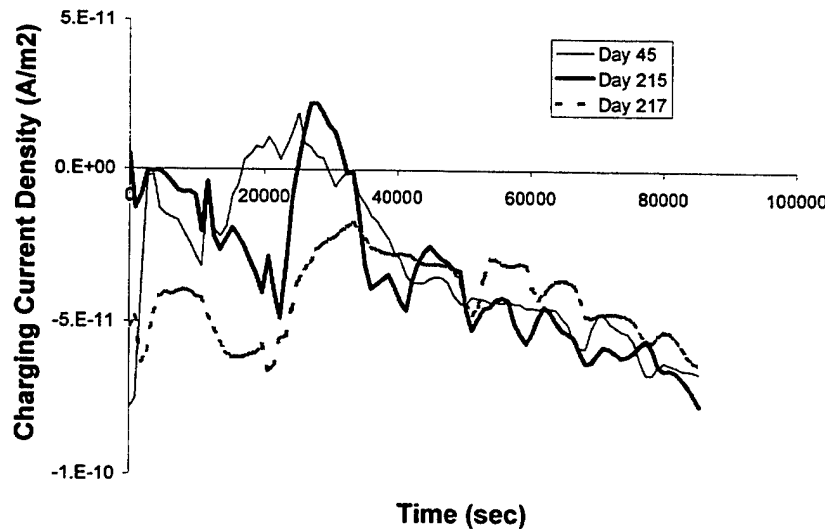


Figure 17. Calculated net charging current densities for the three chosen days. Spacecraft charging is predicted for a brief period on two of the days.

The net charging current is never more than a few percent of the incident electron current due to secondary, backscatter, and ion currents. Figure 18 shows the contributions to the current on day 215, 1996. The net charging current is less than 10% of the incident electron current. All the

other contributions to the net current (secondary and backscattered electrons, and incident ions) are of the opposite sign from the incident electron current.

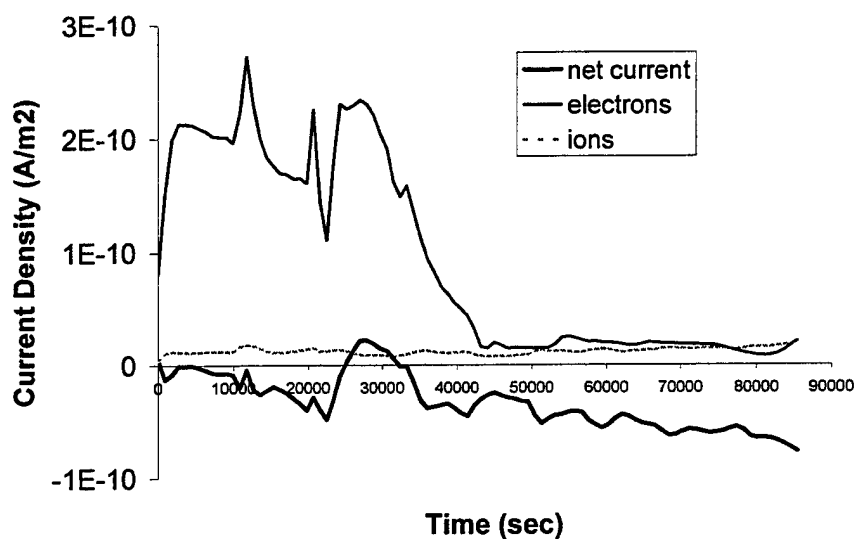


Figure 18. Incident electron and ion, and the net charging current density as calculated for day 215, 1996.

7. References

- ¹ S. A. Brebbia, *Boundary Element Methods*, Springer Verlag, New York, 1981.
- ² S. M. Selby, *Standard Mathematical Tables*, CRC Press, Cleveland, OH, 1975
- ³ M. J. Mandell, I. Katz, G. W. Schnuelle, P. G. Steen, J. C. Roche, "The Decrease in Effective Photocurrents due to Saddle Points in Electrostatic Potentials near Differentially Charged Spacecraft," *IEEE Trans. Nucl. Sci. NS-25*, 1978.
- ⁴ I. Katz, V. A. Davis, M. J. Mandell, B. M. Gardner, J. M. Hilton, J. Minor, , A. R. Fredrickson, D. L. Cooke, *Interactive Spacecraft Charging Interactive Handbook with Integrated Updated Spacecraft Charging Models*, AIAA paper 2000-0247, Presented at the Aerospace Sciences Meeting, Jan. 2000.